

# **Exhibit 16**

## **Part 2**



## OneJava Development Framework Features

- Dynamic invocation of services (Inversion of Control container?)
- Remoting support to seamlessly work in an N-tier architecture (e.g. Adobe BlazeDS, .NET RIA Services)
- Messaging framework that allows data sharing among application components and also among different applications
- Extensibility for adding or removing new platform services
- Application lifecycle events
- L10N & I18N
- Accessibility
- Offline applications support



## Why a New Development Framework?

- Extend the Java platform to 8 Million+ web developers
- Make it easy to build and deploy N-tier enterprise mobile applications on Java
- Enable carriers and operators to quickly and easily offer new services to consumers
- Provide a simple development environment that allows an efficient, iterative designer-developer workflow
- Allow applications to leverage the vast amount of web content



# Deployment Considerations

- OneJava should have a common application packaging format for all device platforms
- Allow dynamic loading of libraries (modularization?)
- Packaging format should support deployment
  - From a provisioning service such as Java Marketplace or CDS
  - Downloading from a network location
  - Directly from a computer tethered to a phone
  - Transfer from one phone to another via Bluetooth
- Should support remote and background loading of assets and libraries from network
- Support a security policy
- Provide an application launch experience that is highly customizable and user-friendly



# HTML5 Top Features

- Web workers – Background threads for processing
- <Video> and <Audio>
- <Canvas> for graphics
- Storage API for offline applications
- Geolocation
- New structural elements - <article>, <footer>
- Easy web forms validation
- Contenteditable attribute to mark regions for user-generated content





# HTML5 Advantages & Disadvantages

## Advantages

- Minimal learning curve, large pool of HTML/CSS developers
- Plenty of free HTML WYSIWYG tools
- No plug-in needed, runs on any browser
- Search engine friendly



## Disadvantages

- W3C standards body moves very slowly
- User experience will vary between browser/hardware platforms
- HTML apps will be constrained to the capabilities of the browser
  - Limited support for media formats
  - Lack of good tooling for developing creative HTML 5 applications



# OneJava Market Landscape Discussion

CSCG Marketing  
Noel Poore





# Trends

- Content is a mashup of RIA, Web and Data Management
  - ▲ Immersive user experience
  - ▲ Data driven from web services
  - ▲ Managed locally on device through computational, business logic
  - ▲ Competition enabling all content types
- Java is perceived as stagnant and legacy
  - ▲ Stagnant innovation
  - ▲ Only aimed at Java programmers
  - ▲ Fragmented between Java SE and Java ME, and between Java ME Mobile and TV and within Mobile and TV



# Competition enabling new models

- Android, ChromeOS, Flash, HTML5/JavaScript
- Multiple development models and developer types
- Seamless end-to-end app platforms
- Simplified and assisted development/deployment (tools, stores, existing back-end services, etc)
- Single runtime for the multiple app models
- Minimal differentiation across enterprise or consumer apps



# Modernize and Unify Java

- Leverage strengths of Java:
  - Portability and ubiquity
  - Existing standardized, implemented device and network APIs
  - A true end-to-end platform
  - Existing investments from ecosystem
- Define one development model across all device types, allowing for industry-specific customization w/o fragmentation of features
  - “Common-izing” Java
  - Expanding content developer audience to Web developers and designers
  - Blended Apps
- Simplify development and deployment of content through Sun tools, store and services
- Run Web, RIA and Java content on the Java runtime

2009 JavaOne VIP Program, San Francisco

Slide 4

Sun Microsystems Proprietary - Confidential

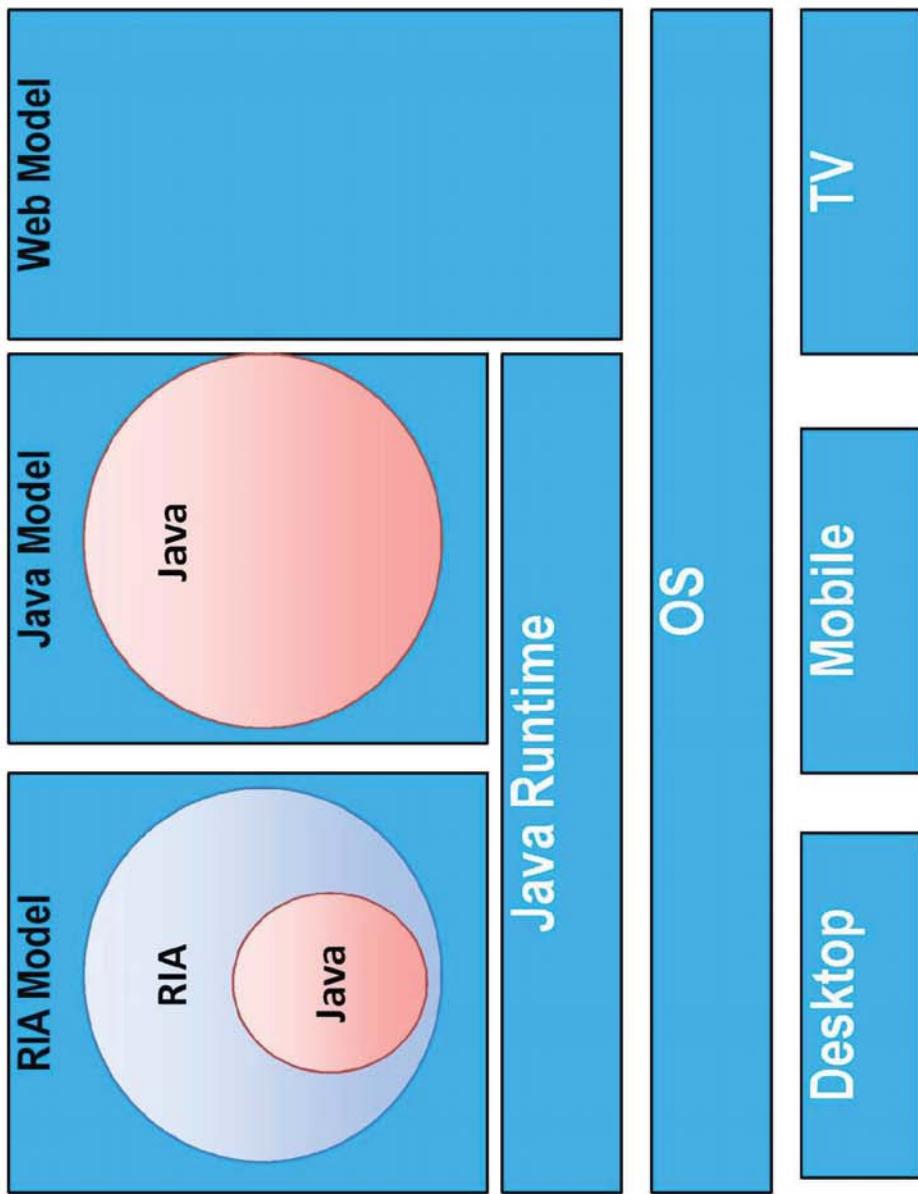
CONFIDENTIAL

Trial Exhibit 3508 Page 34 of 58

OAGOOGLE00008486529



# The App Platform model today



2009 JavaOne VIP Program, San Francisco

Slide 5

Sun Microsystems Proprietary - Confidential

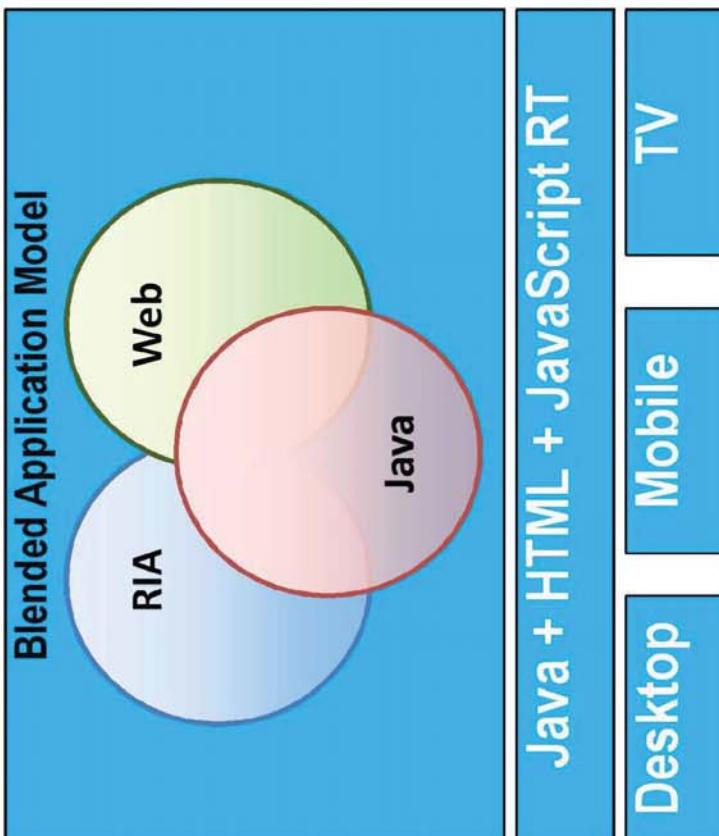
CONFIDENTIAL

Trial Exhibit 3508 Page 35 of 58

OAGOOGLE00008486530



## Key Advantages of OneJava



- Incumbent, standardized, cross-screen, open ties to device capabilities
- Extremely fast VM
- Large developer base
- Protecting ecosystem investment
- Ubiquity as opposed to siloed platforms (Android, iPhone)
- Addresses fragmentation

2009 JavaOne VIP Program, San Francisco

Slide 6

Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 36 of 58

OAGOOGLE00008486531



## Target Device Segments

- Smartphones (read: Open OS-based phones)
  - Next growth in wireless space
- Netbooks / MIDs
  - Likely the fastest growing consumer device for next 2-3 years
- Hi-end TV / Set Top Boxes
  - Sets up for Quad play
- Desktop / Laptops
  - Current battleground
- Minimum hardware: T-Mobile G2 class hardware
- Potentially any other devices that fits min HW



## OneJava is NOT....

- Not a software stack for mobile, DTV devices or anything else (No OS, apps, device specific frameworks included)
- Not a solution w/ bundled applications
- Not a solution w/ a \*top (phonetop, TV-top)
- Not a replacement for Java ME in feature phones, Java ME in TV



## Feature Quick Hits for OneJava 1.0

- Hotspot VM and core libraries from JDK7 codebase
- SE subset APIs + ME APIs + extras + cleanup
- Initial device targets: G2 handset (Android) and Acer Netbook (Linux)
- JavaScript engine integrated
- JavaFX 1.3 (SoMa) integrated
- HTML5/CSS integrated
- Use JavaFX model for common & profile definition
- Embedded runtime expectations of performance and memory usage



## Monetization of OneJava

- OneJava is a platform: an application model, a runtime and a set of tools
  - Not a business
- CSG traditional businesses (desktop, mobile and TV) will implement products and services based on OneJava
  - Industry specific
  - Potentially different business models
  - TB discussed later
- Example: OneJava for Mobile is likely a Smartphone product w/ a set of Carrier Services, and monetization happens at carrier

2009 JavaOne VIP Program, San Francisco

Slide 10

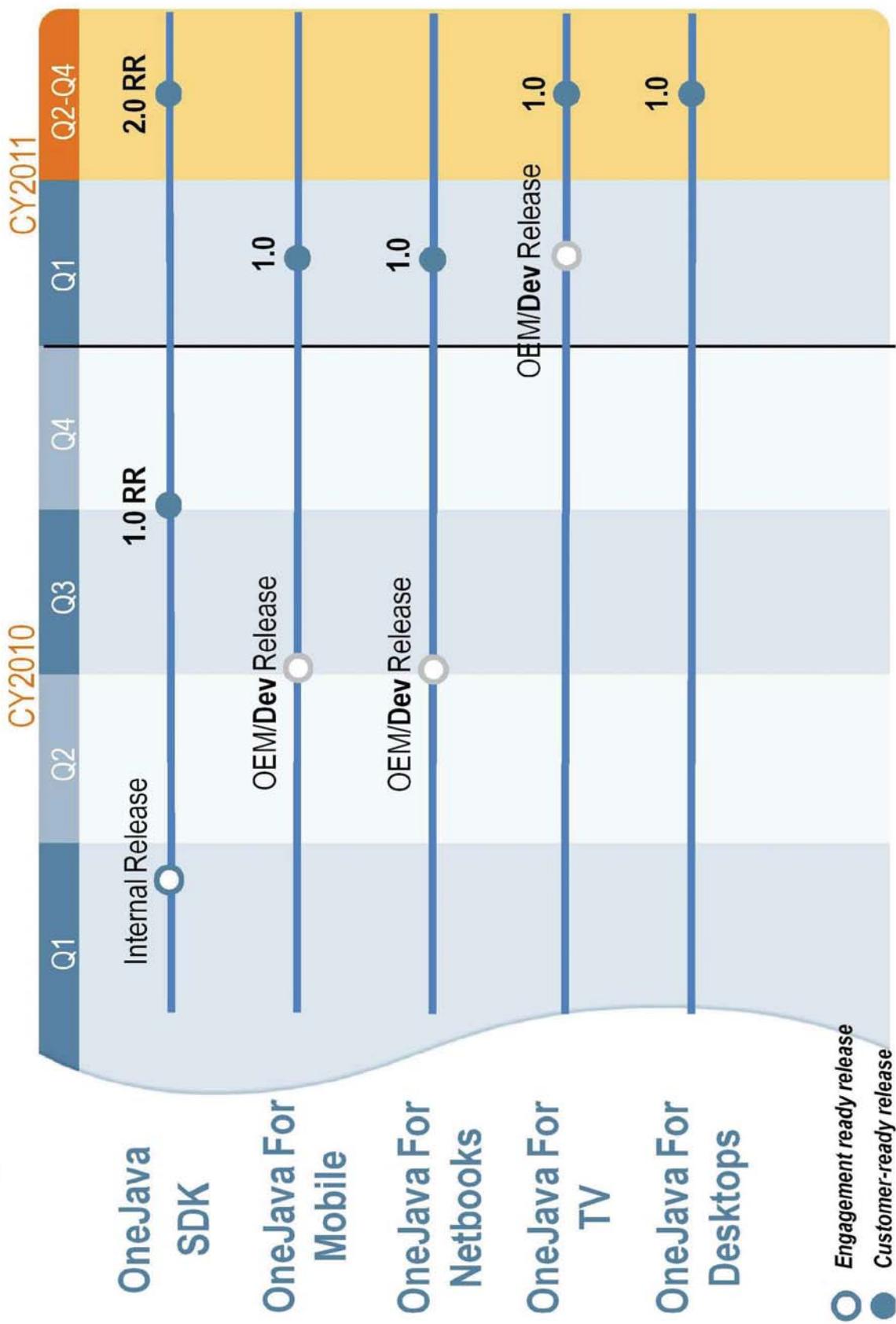
Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 40 of 58

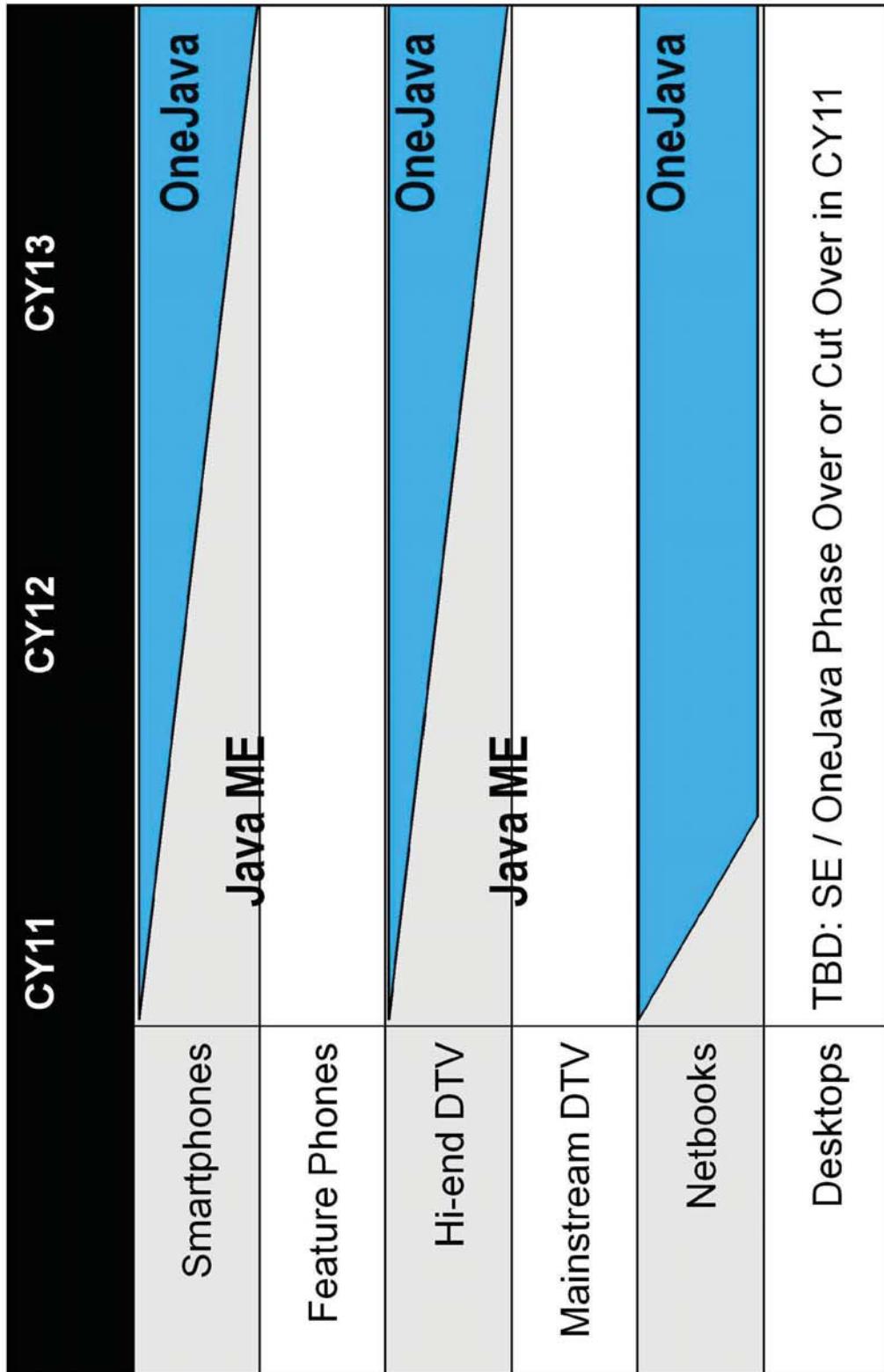
OAGOOGLE00008486535

# Proposed Roadmap





# Expected Adoption of OneJava



2009 JavaOne VIP Program, San Francisco

Slide 12

Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 42 of 58

OAGOOGLE00008486537



# OneJava Developer Model Discussion

Jai Suri  
Noel Poore



Sun Microsystems Proprietary - Confidential

CONFIDENTIAL



# Development & Deployment Environment

## Tools

- IDE
- RAD Tool
- Authoring Tool

## Command Line Tools

- Packaging
- Project Creation
- Debugging

## SDK Documentation

## Tool Extensions

- OneJava SDK
- Emulator integration
- On-Device debugging
- Messages console
- Applications log
- Assets Importer
- Services Integration

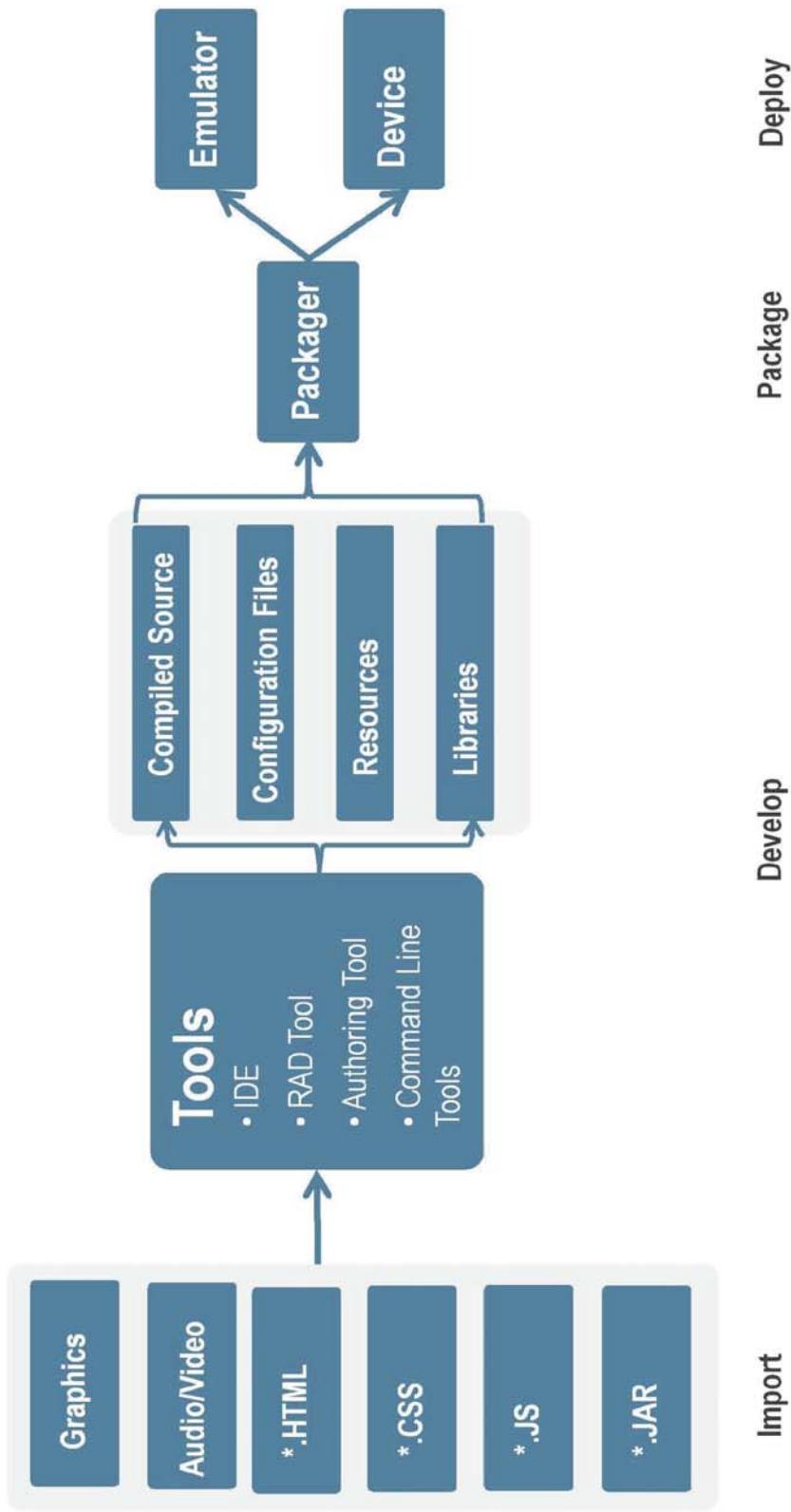
## Emulator

- Skins/Profiles
- Network Simulation

## Development Device

## Server-side Integration Framework

# Development Workflow



2009 JavaOne VIP Program, San Francisco

Slide 15

Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 45 of 58

OAGOOGLE00008486540



# Developer Profiles

## Sample profiles of developers building applications on OneJava

	Web Developer	Mobile (Java) Developer	JavaFX Developer	Java SE Developer
Which programming languages will they use?	HTML, JavaScript, CSS	Java	JavaFX Script, CSS, Java	Java
What tools will they use?	Authoring Tool RAD Tool	RAD Tool IDE	Authoring Tool RAD Tool	RAD Tool IDE
What types of applications will they build?	End-user applications	<ul style="list-style-type: none"> <li>• End-user applications</li> <li>• On-device services used by applications</li> </ul>	End-user applications	<ul style="list-style-type: none"> <li>• End-user applications</li> <li>• On-device services used by applications</li> </ul>
What frameworks will they use?	JavaScript/AJAX frameworks e.g., JQuery	Custom MVC framework Legacy support to run LWUIT, MSA applications	Custom MVC framework	Custom MVC framework



# Development Model

## Considerations

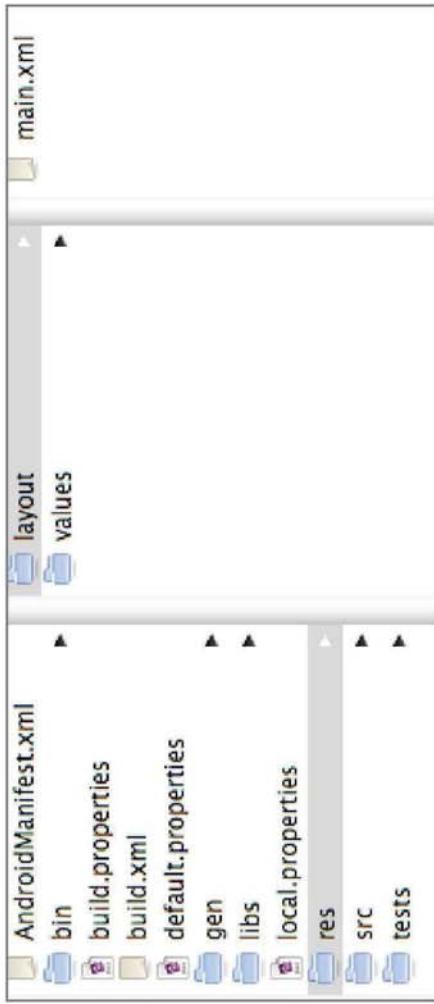
- OneJava is primarily a client-side platform
- Hence simplifying UI development is paramount to success
- Typical UI consists of various components
  - Data
  - Interaction design
  - Visual design
  - Connectivity
  - Multi-threading
  - Security
  - I18N
  - Validation
  - Remoting
  - Unit testing
  - Multimedia
  - Black magic
- Easily fit into an N-tiered service-oriented architecture



# Competitive Development Frameworks

Most UI technologies have adopted MVC or derivative framework to simplify UI development, including new mobile platforms such as Android and Palm WebOS

## Android's Application Framework



```
$ android create project \
--package com.android.helloandroid \
--activity HelloAndroid -target 2 \
--path <path-to-your-project>/HelloAndroid
```

2009 JavaOne VIP Program, San Francisco

Slide 18

## WebOS' Application Framework



```
$ palm-generate AppName
```

Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 48 of 58

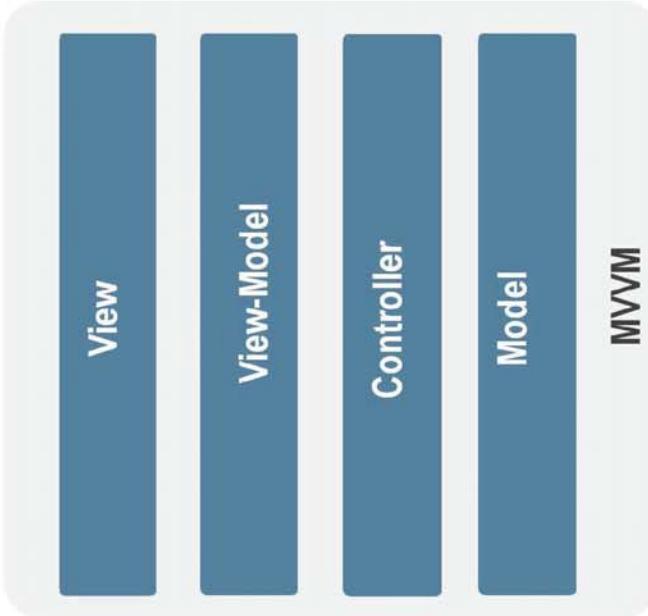
OAGOOGLE00008486543



# Development Framework Guidelines

## OneJava Application Framework characteristics

- Ideal framework would be MVVM (Model-View-ViewModel)
  - Separates views from view logic enabling cross-screen UI development
- Framework should not be mandatory to develop applications
- Framework should scale from none -> MVC -> MVVM
- Should enable easy service-oriented application development





# Development Framework Guidelines

## OneJava Application Framework characteristics

- > Clearly separates view, model and control layers
- > Allows developers in a team to work independently on each layer and integrate later
- > Allows developers to use the technology that works best for each layer
- > Allows each layer to be unit tested independently

# OneJava Development Frameworks

Current development models on Java platform are

## #1: Java-based Development Model

Java      UI, Interaction, Services, Platform API

## #2: JavaFX-based Development Model

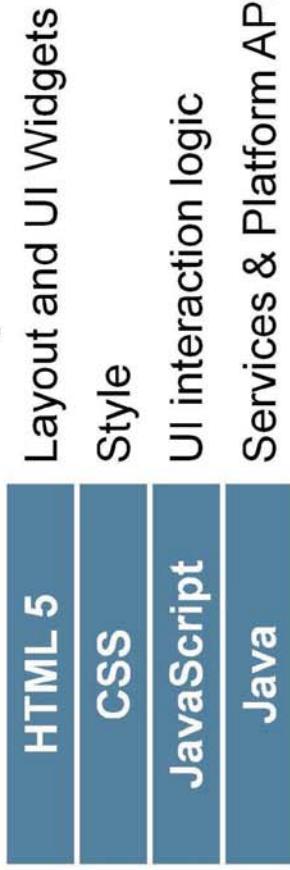
JavaFX Script	UI (Graphics, Rich media, UI Widgets), Interaction
CSS	Style
Java	Services, Platform API



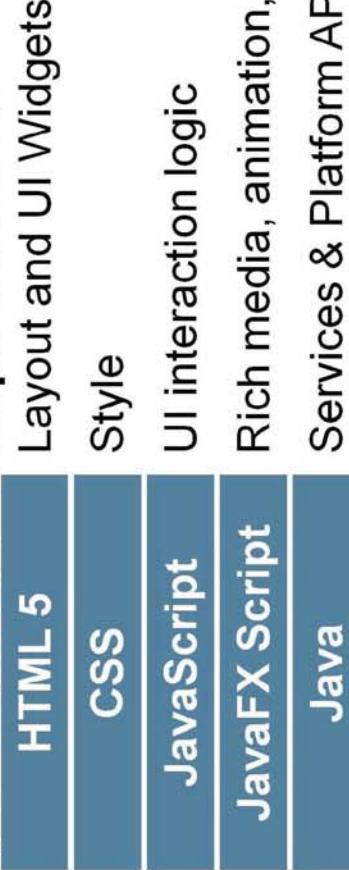
# OneJava Development Frameworks

OneJava Development framework should offer web application development models that allow developers to combine the strengths of web technologies

## #3: Web-based Development Model



## #4: Blended Development Model



2009 JavaOne VIP Program, San Francisco

Slide 22

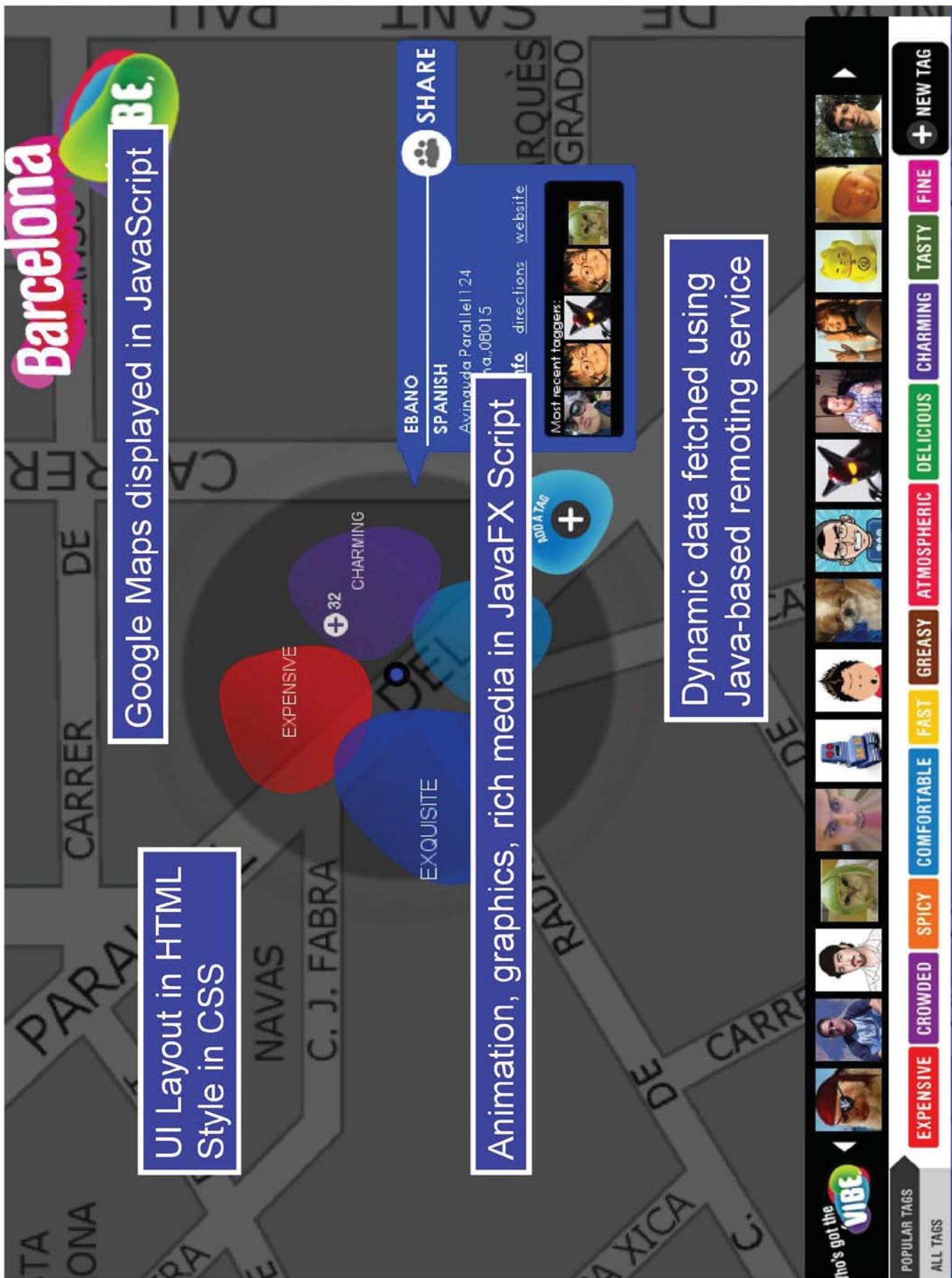
Sun Microsystems Proprietary - Confidential

CONFIDENTIAL

Trial Exhibit 3508 Page 52 of 58



OAGOOGLE00008486547





## OneJava Development Framework Features

- Dynamic invocation of services (Inversion of Control container?)
- Remoting support to seamlessly work in an N-tier architecture (e.g. Adobe BlazeDS, .NET RIA Services)
- Messaging framework that allows data sharing among application components and also among different applications
- Extensibility for adding or removing new platform services
- Application lifecycle events
- L10N & I18N
- Accessibility
- Offline applications support



## Why a New Development Framework?

- Extend the Java platform to 8 Million+ web developers
- Make it easy to build and deploy N-tier enterprise mobile applications on Java
- Enable carriers and operators to quickly and easily offer new services to consumers
- Provide a simple development environment that allows an efficient, iterative designer-developer workflow
- Allow applications to leverage the vast amount of web content



# Deployment Considerations

- OneJava should have a common application packaging format for all device platforms
- Allow dynamic loading of libraries (modularization?)
- Packaging format should support deployment
  - From a provisioning service such as Java Marketplace or CDS
  - Downloading from a network location
  - Directly from a computer tethered to a phone
  - Transfer from one phone to another via Bluetooth
- Should support remote and background loading of assets and libraries from network
- Support a security policy
- Provide an application launch experience that is highly customizable and user-friendly



# HTML5 Top Features

- Web workers – Background threads for processing
- <Video> and <Audio>
- <Canvas> for graphics
- Storage API for offline applications
- Geolocation
- New structural elements - <article>, <footer>
- Easy web forms validation
- Contenteditable attribute to mark regions for user-generated content





# HTML5 Advantages & Disadvantages

## Advantages

- Minimal learning curve, large pool of HTML/CSS developers
- Plenty of free HTML WYSIWYG tools
- No plug-in needed, runs on any browser
- Search engine friendly



## Disadvantages

- W3C standards body moves very slowly
- User experience will vary between browser/hardware platforms
- HTML apps will be constrained to the capabilities of the browser
  - Limited support for media formats
  - Lack of good tooling for developing creative HTML 5 applications